Electronic Version
Stylesheet Version v1.1.1

Claims

[c1] What is claimed is:

1. A digital video (DV) storage system comprising:
an interface module receiving an incoming signal and converting the incoming signal into an incoming bit-stream;
a DV demuxer directly connected to the interface module for receiving the incoming bit-stream, wherein the DV demuxer de-multiplexes received blocks in the incoming bit-stream into at least video blocks being in video sections and audio blocks being in audio sections; and
a memory coupled to the DV demuxer for storing the video blocks and audio blocks;
wherein the incoming bit-stream is not buffered outside the interface module and the DV demuxer.

[c2] 2. The DV storage system of claim 1, wherein the interface module is an IEEE 1394 interface module.

[c3] 3. The DV storage system of claim 1, wherein the DV demuxer further manages a write block pointer and determines if the incoming bit-stream is compliant with a DV format.

[c4] 4. The DV storage system of claim 3, wherein the DV demuxer comprises:
a data extractor receiving the incoming bit-stream and checking the incoming bit-stream for errors to determine if the incoming bit-stream is compliant with the DV format before de-multiplexing the incoming bit-stream into the video and audio blocks; and
a buffer manager having a memory interface coupled to the memory, the buffer manager storing the video and audio blocks in the memory using the memory interface according to the write block pointer.

[c5] 5. The DV storage system of claim 4, wherein the DV demuxer comprises a host controller, and the data extractor outputs received blocks of sections other than the video and audio sections to the host controller.

[c6] 6. The DV storage system of claim 4, wherein the incoming signal contains packets and the interface module outputs a packet start indication to indicate the beginning of each packet in the incoming bit stream; and
the data extractor compares the number of double words received in the incoming bit stream starting at the packet start indication with a first predetermined value, the data extractor determining the incoming bit-stream to have an error if the number of double words received exceeds the first predetermined value.

[c7] 7. The DV storage system of claim 4, wherein the data extractor compares a received block number order of the received blocks in the incoming bit-stream with a predetermined order, the data extractor determining the incoming bit-stream to have an error if the received block number order differs from the predetermined order.

[c8] 8. The DV storage system of claim 4, wherein the data extractor compares a received sequence number order of the received blocks in the incoming bit-stream with a predetermined order, the data extractor determining the incoming bit-stream to have an error if the received sequence number order differs from the predetermined order.

[c9] 9. The DV storage system of claim 4, wherein the memory manager sequentially stores the video and audio blocks in respective sections of the memory according to the write block pointer;
wherein if the data extractor determines the incoming bit stream to have an error, the memory manger returns to the beginning of the respective sections.

[c10] 10. The DV storage system of claim 4, wherein the memory manager sequentially stores the video and audio blocks in respective sections of the memory according to

the write block pointer;
wherein if the data extractor determines the incoming bit stream to have an error, the memory manger increments the write block pointer and skips to the beginning of the respective sections according to the incremented write block pointer.

[c11] 11. The DV storage system of claim 4, wherein the memory manager stores the video and audio blocks in respective sections of the memory, the respective sections of the memory being determined according to the write block pointer; the video and audio blocks being stored within the respective sections according to a sequence number and a block number of each video and audio block in the incoming bit-stream.

[c12] 12. A method of storing digital video (DV) data, the method comprising the following steps:
providing an interface module for receiving an incoming signal and converting the incoming signal into an incoming bit-stream;
directly receiving the incoming bit-stream from the interface module;
de-multiplexing received blocks in the incoming bit-stream into at least video blocks being in video sections and audio blocks being in audio sections; and
storing the video blocks and audio blocks in a memory.

[c13] 13. The method of claim 12, wherein the interface module is an IEEE 1394 interface module.

[c14] 14. The method of claim 12, further comprising:
determining if the incoming bit-stream is compliant with a DV format; and
managing a write block pointer;
wherein the step of storing the video blocks and audio blocks in a memory is performed according to the write block pointer.

[c15] 15. The DV storage system of claim 14, further comprising providing a DV demuxer directly connected to the interface module having no buffer or memory between the interface module and the DV demuxer; wherein the steps of receiving the incoming bit-stream, determining if the incoming bit-stream is compliant with the DV format, de-multiplexing the received blocks in the incoming bit-stream into at least the video blocks being in video sections and the audio blocks being in audio sections; and managing the write block pointer are performed by the DV demuxer.

[c16] 16. The method of claim 14, further comprising providing a DV demuxer directly connected to the interface module; wherein the method further comprises:

utilizing the DV demuxer to receive the incoming bit-stream and check the incoming bit-stream for errors to determine if the incoming bit-stream is compliant with the DV format before de-multiplexing the incoming bit-stream into the video and audio blocks; and
utilizing the DV demuxer to store the video and audio blocks in the memory using a memory interface according to the write block pointer.

[c17] 17. The method of claim 16, further comprising providing a host controller; wherein the method further comprises outputting received blocks of sections other than the video and audio sections to the host controller.

[c18] 18. The method of claim 16, wherein the incoming signal contains packets and the interface module outputs a packet start indication to indicate the beginning of each packet in the incoming bit stream; the method further comprising:
utilizing the DV demuxer to compare the number of double words received in the incoming bit stream starting at the packet start indication with a first predetermined value, and determining the incoming bit-stream to have an error if the number of double words received exceeds the first predetermined value.

[c19] 19. The method of claim 16, further comprising utilizing

the DV demuxer to compare a received block number order of the received blocks in the incoming bit-stream with a predetermined order, and determining the incoming bit-stream to have an error if the received block number order differs from the predetermined order.

[c20] 20. The method of claim 16, further comprising utilizing the DV demuxer to compare a received sequence number order of the received blocks in the incoming bit-stream with a predetermined order, and determining the incoming bit-stream to have an error if the received sequence number order differs from the predetermined order.

[c21] 21. The method of claim 16, further comprising:
sequentially storing the video and audio blocks in respective sections of the memory according to the write block pointer; and
if the data extractor determines the incoming bit stream to have an error, returning to the beginning of the respective sections of the memory.

[c22] 22. The method of claim 16, further comprising:
sequentially stores the video and audio blocks in respective sections of the memory according to the write block pointer; and
if the data extractor determines the incoming bit stream

to have an error, incrementing the write block pointer and skipping to the beginning of the respective sections according to the incremented write block pointer.

[c23] 23. The method of claim 16, further comprising:
storing the video and audio blocks in respective sections of the memory, the respective sections of the memory being determined according to the write block pointer; and
storing the video and audio blocks within the respective sections according to a sequence number and a block number of each video and audio block in the incoming bit-stream.